Computational comparisons of different formulations for the Bilevel Minimum Spanning Tree Problem

Martine Labbé, Miguel Pozo, Justo Puerto

xx (2018) 1–19

Computational comparisons of different formulations for the Bilevel Minimum Spanning Tree Problem

Martine Labbé[a,b], Miguel A. Pozo[c,*] and Justo Puerto[d]

[a]*Computer Science department. Universit Libre de Bruxelles, Faculty of Sciences, Boulevard du Triomphe CP 210 / 01, B-1050 Brussels, Belgium.*
[b]*INOCS Team, INRIA, 165 Avenue de Bretagne, 59000 Lille France.*
[c]*Department of Applied Math II, University of Seville, Escuela Tcnica Superior de Ingeniera, Camino de los Descubrimientos s/n, Sevilla 41092, Spain.*
[d]*Department of Statistics and Operational Research, University of Seville, Facultad de Matemticas. C/ Tarfia s/n, Sevilla 41012, Spain.*
E-mail: mlabbe@ulb.ac.be [Martine Labbé]; miguelpozo@us.es [Miguel A. Pozo]; puerto@us.es [Justo Puerto]

Abstract

Let be given a graph $G = (V, E)$ whose edge set is partitioned into a set R of red edges and a set B of blue edges, and assume that red edges are weighted and contain a spanning tree of G. Then, the Bilevel Minimum Spanning Tree Problem (BMSTP) is that of pricing (i.e., weighting) the blue edges in such a way that the total weight of the blue edges selected in a minimum spanning tree of the resulting graph is maximized. In this paper we present different mathematical formulations for the BMSTP based on the properties of the Minimum Spanning Tree Problem and the bilevel optimization. We establish a theoretical and empirical comparison between these new formulations and we also provide reinforcements that together with a proper formulation are able to solve medium to big size instances random instances. We also test our models in instances already existing in the literature.

Keywords: Minimum Spanning Tree; Bilevel optimization; Stackelberg game

1. Introduction

Let G be a given a graph whose edge set is partitioned into a set of red edges and a set of blue edges, and assume that red edges are weighted and contain a spanning subgraph of G. Then, the Bilevel Minimum Spanning Tree Problem (BMSTP) consists in pricing (i.e., weighting) the blue edges in such a way that the total weight of the blue edges selected in a Minimum Spanning Tree (MST) of the resulting graph is maximized.

*e-mail: miguelpozo@us.es.

An example of the BMSTP is the following. Suppose a telecommunications company (TC) owns several connections (blue edges) between different nodes of a network. A new provider wants to enter into the market building a network that connects all nodes at minimum cost. It may use the connections of TC or those of its competitors (red edges). The target is to maximize the profit of the connections that TC could sell to the new provider.

The BMSTP can be seen as a bilevel optimization problem where the second level is a MST and where the objective functions are bilinear at both levels. Optimization problems related to spanning trees, or simply Spanning Tree Problems (STP), are among the core problems in combinatorial optimization. On the one hand, the combinatorial object that represents spanning trees has important structural properties. On the other hand, from a practitioner point of view, spanning trees are found in a wide range of applications in many fields (e.g. computer networks design, telecommunications networks, transportation, etc). Furthermore, they often appear as subproblems of other more complex optimization problems.

In game theory, a bilevel optimization problem is known under the name of Stackelberg game (von Stackelberg, 1934) and it consists in a game between a leader and a follower who play sequentially. Those players compete with each other: the leader makes the first move, and then the follower reacts optimally to the leader's action. This kind of hierarchical game is asymmetric in nature, where the leader and the follower cannot be interchanged. The leader knows ex ante that the follower observes its actions before responding in an optimal manner. Therefore, to optimize its objective, the leader anticipates the optimal response of the follower. In this setting, the leader's optimization problem contains a nested optimization task that corresponds to the follower's optimization problem.

Several papers have been published covering the problem in which the second level consists in choosing shortest paths between pairs of origins and destinations. The problem was first introduced by Labbé et al. (1998). Roch et al., 2005 show that the problem is strongly NP-hard even when the second level consists in one single shortest path. More references regarding that bilevel optimization problem can be found in the surveys of van Hoesel (2008) and Labbé and Violin (2013).

Gassner (2002) studied a discrete variant of the bilevel minimum spanning tree problem where a partition of the set of edges into leader- and follower-edges is given. The leader's action is to choose a subset of his edges while the follower's reaction is to build up a spanning tree that includes the edges chosen by the leader. Hence, the leader's and follower's decision vectors are discrete.

Cardinal et al. (2011) proved the APX-hardness of the BMSTP even when the number of red edge costs is 2, and gave an approximation algorithm with guaranteed worst case performance. They also give an integer programming formulation for the problem and study its linear programming relaxation. Further Cardinal et al. (2013) proved that the problem remains NP-hard even if G is planar, while it can be solved in polynomial time once that G has bounded treewidth.

Bilò et al. (2015) point out that the hardness in finding an optimal solution for the BMSTP lies in the selection of the optimal set of blue edges that will be purchased by the follower. Since once a set of blue edges is part of the final MST, their best possible pricing can be computed in polynomial time, as shown in Cardinal et al. (2011).

Morais et al. (2016) introduce a reformulation and a Branch-and-cut-and-price algorithm for BMSTP. The reformulation is obtained after applying KKT optimality conditions to a BMSTP non-compact Bilevel Linear Programming formulation and is strengthened with a partial rank-1 RLT and with valid inequalities from the literature. They also implemented a Branch-and-cut algorithm for an extended formulation derived from another one in the literature and a preliminary computational study comparing both methods is presented.

In this paper we will present different mathematical formulations for the BMSTP based on the properties of the MSTP and the bilevel optimization paradigm. We establish a theoretical and empirical comparison between these new formulations that are able to solve medium to big size random instances. We also test our models in instances already existing in the literature taken from Morais et al. (2016).

The remainder of the paper is organized as follows. In Section 2 we formally define BMSTP and provide a new heuristic algorithm. Sections 3 and 4 present the catalogue of formulations that we study for the BMSTP including those corresponding to the MST subproblem. The empirical performance of the resulting BMSTP formulations is analyzed in Section 5, where we present extensive numerical results and a comparison with existing ones. Finally, some conclusions are summarized in Section 6.

2. Problem description and preliminary results

The BMSTP can be formally defined as follows. Let be given a graph $G = (V, E)$ whose edge set E is partitioned into a set B of blue edges (controlled by the leader) and a set R of red edges, and assume that red edges are weighted and contain at least one spanning tree of G, thus, $|R| \geq |V| - 1$. A positive cost c_e is associated to each red edge $e \in R$ and a positive price T_e ($T = [T_1, ..., T_{|E|}]$) has to be determined for each blue edge $e \in B$. We denote by $x = [x_1, ..., x_{|E|}]$ the design variables used to describe the spanning tree polytope $\mathcal{T}$.

Then, the Bilevel Minimum Spanning Tree Problem (BMSTP) consists in determining T_e for each $e \in B$ in such a way that the total weight of the blue edges selected in a minimum spanning tree of the resulting graph is maximized. A very general non-linear model for the BMSTP is then the following:

$$\text{BMSTP}: \quad \max_{T \geq 0} \sum_{e \in B} T_e x_e \tag{1a}$$

$$\text{s.t.:} \quad x = \operatorname*{argmin}_{x \in \mathcal{T}} \{ \sum_{e \in B} T_e x_e + \sum_{e \in R} c_e x_e \}. \tag{1b}$$

Objective function (1a) maximizes the total weight of the blue edges selected in the solution. Constraints (1b) return the MST in the graph for a given cost vector T.

Example 1. Let G be a graph as depicted in Figure 1 left where red edges provide a spanning tree of total cost 20. Blue edges can be priced as in Figure 1 center or Figure 1 right in order to provide a BMSTP solution of value 15.

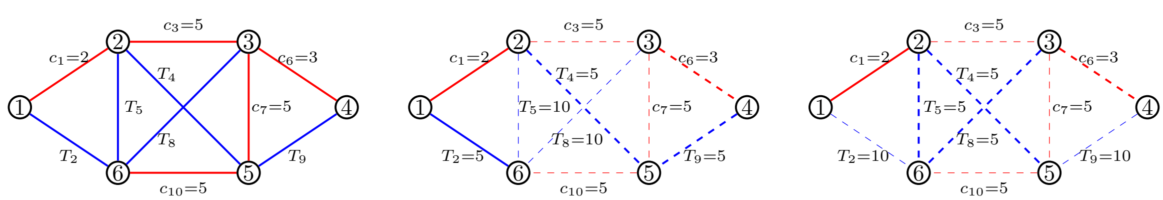

Fig. 1. Graph with edge costs (left) and two BMST optimal solutions for $|B| = |E| - (|V| - 1)$

2.1. Basic results

We first observe (see Cardinal et al., 2013) that in every optimal BMSTP solution the prices T_e, $e \in B$ take values in the set of red edges cost.

Property 1. *In every optimal BMSTP solution* $T_e \in C^R = \{c_{e'} : e' \in R\}$ *for each* $e \in B$.

Second, we observe (see Cardinal et al., 2013) that the cost of each blue edge belonging to an optimal solution is bounded from above by the minimum among the maximum of the red costs of each cycle that contain the blue edge. Let $\mathcal{C}(e, S)$ be the set of cycles of G that include edge e and edges of the set $S \subseteq E \backslash \{e\}$. Then,

Property 2. *(*Strong necessary condition for optimal BMSTP solution*) If* T^* *is an optimal BMSTP solution and* $\mathcal{T}^*$ *the associated tree, then*

$$T_e \leq \min_{\Theta \in \mathcal{C}(e,E)} \max_{e' \in \Theta \cap R} c_{e'}, \;\; e \in B \cap \mathcal{T}^*$$

In particular, we observe that each blue edge in an optimal solution verifies that its price is upper bounded by the maximum cost of red edges for which belongs to a path linking the end vertices of the blue edge and containing only red edges. Analogously, the cost of each blue edge of an optimal spanning tree is bounded from below by the minimum cost of a red edge that belongs to a path linking the end vertices of the blue edge and containing only red edges.

Property 3. *(*Weak necessary condition for optimal BMSTP solution*) If* T *is an optimal BMSTP solution then*

$$m_e = \min_{\Theta \in \mathcal{C}(e,R), e' \in \Theta} c_{e'} \leq T_e \leq \max_{\Theta \in \mathcal{C}(e,R), e' \in \Theta} c_{e'} = M_e \;\; e \in B$$

Property 3 provides upper and lower bounds for each variable T_e, $e \in B$.

It is known that a spanning tree $\mathcal{T}^*$ is an optimal MSTP solution if for each edge $e \notin T^*$, each edge $e' \in T^*$ and in the cycle that contains e has a cost less or equal than c_e, that is

$$\mathcal{T}^* \text{ is an optimal MSTP solution } \Leftrightarrow c_{e'} \leq c_e, \forall e \in E : e \notin \mathcal{T}^*, e' \in \mathcal{C}(e, \mathcal{T}^*) : e' \neq e$$

Similarly, depending if e, e' belong to R or B we can provide an optimality condition for the BMSTP:

Property 4. *(*BMSTP optimality condition*) If* $\mathcal{T}^*$ *is the associated tree of an optimal BMSTP solution then:*

$$c_{e'} \leq c_e \qquad e \in R : e \notin \mathcal{T}^*, e' \in R \cap \mathcal{C}(e, \mathcal{T}^*) : e' \neq e \tag{2a}$$
$$T_{e'} \leq c_e \qquad e \in R : e \notin \mathcal{T}^*, e' \in B \cap \mathcal{C}(e, \mathcal{T}^*) : e' \neq e \tag{2b}$$
$$c_{e'} \leq T_e \qquad e \in B : e \notin \mathcal{T}^*, e' \in R \cap \mathcal{C}(e, \mathcal{T}^*) : e' \neq e \tag{2c}$$
$$T_{e'} \leq T_e \qquad e \in B : e \notin \mathcal{T}^*, e' \in B \cap \mathcal{C}(e, \mathcal{T}^*) : e' \neq e \tag{2d}$$

Cardinal et al. (2011) provides a simple approximation algorithm, called *Best-Out-Of-k algorithm*: Let $c^1 < ... < c^{|K|}$ be the $|K|$ different edge costs that appear in the initial red set of edges, where $k \in K$ is the index of the $k-th$ cost and $C^R = \{c^1 < ... < c^{|K|}\}$ is the set of costs. The Best-Out-Of-k algorithm consists in choosing the best cost c^k to be assigned to all T values. We can also rewrite this algorithm as a property in the following way:

Property 5. *(*BMSTP feasibility*) A feasible BMSTP solution (lower bound) is given by:*

$$T = (c^k)_{1\times|B|} \ / \ k = \arg\max_{k\in K} \left\{ \sum_{e\in B} c^k x_e : x = \underset{x\in\mathcal{T}}{\operatorname{argmin}} \{ \sum_{e\in B} c^k x_e + \sum_{e\in R} c_e x_e \} \right\}$$

2.2. A general framework for providing BMSTP feasible solutions

Algorithm 1: BMSTP-H algorithm

input :
- sol_{best}: Current best solution (by default $T_e = c^{|K|}, \forall e \in B$).
- b: Number of blue edges to modify (by default $b = |B|$)
- S: Set of edges that has been chosen in previous iterations (by default $S = \emptyset$)
- p_1: probability of choosing edges from B or from $B \backslash S$ (by default $p_1 = 0$)
- p_2: probability of choosing a direction of movement where "moving up" is chosen with probability p_2 and "moving down" with probability $1 - p_2$ (by default $p_2 = 0$)
- $STOP_c$: STOP condition (by default "repeat $|K|$ times")

output: sol_{best}: Current best solution.

```
while STOP_c = false do
    According to p_1, a subset B_S of b blue edges is chosen from B ∪ S or from B\S.
    S ← S ∪ B_S.
    Edges e ∈ B_S verifying T_e > M_e or T_e < m_e are removed from B_S.
    According to p_2, for each e ∈ B_S increase/decrease by one unit k in c_e^k.
    Evaluate the BMSTP solution updating c_e^k for all e ∈ B_S.
    if sol_best is outperformed then
        Update sol_best
    else
        reset values c_e^k for all e ∈ B_S
```

In this subsection we generalize the Best-Out-Of-k algorithm by means of a local search algorithm that we denote BMSTP-H. Basically, this algorithm starts fixing the prices of the blue edges with a given value (current best solution) and iteratively, a subset of blue edges is chosen and their associated prices are increased/decreased. The BMSTP solution is then evaluated fixing the new prices of the blue edges. If an improvement is achieved, the current best known solution is updated. Otherwise, we reset the modified blue edges prices and we iterate until a stop condition is fulfilled. For a particular value of the search parameters, this algorithm has as a particular case the Best-Out-Of-k algorithm. However, BMSTP-H allows one to intensify the search by varying (1) the number of blue edges prices to modify, (2) increasing or decreasing the prices, (3) choosing or not blue edges already chosen in previous iterations and (4) modifying the stop condition. Additionally the algorithm can be run several times in a row for different values of the search parameters.

Note that the Best-Out-Of-k algorithm is equivalent to
BMSTP-H($b = |B|, p1 = 1, p2 = 0, sol_{best} = \emptyset, STOP_c =$ "repeat $|K|$ times").

In our experiments we apply three runs of BMSTP-H sequentially. The current best solution achieved on each module is used as an input in the following:

1. BMSTP-H($b = |B|; p1 = 1; p2 = 0; STOP_c =$ "repeat $|K|$ times") $\rightarrow sol_1$.
2. BMSTP-H($b = 1; p1 = 0; p2 = 0; sol_{best} = sol_1$; $STOP_c =$ "repeat until the set of chosen edges have size $|B|$") $\rightarrow sol_2$.
3. BMSTP-H($b = 1, p1 = 0, p2 = 1, sol_{best} = sol_2$; $STOP_c =$ "repeat until the set of chosen edges have size $|B|$") $\rightarrow sol_3$.

We report on the performance of solutions found by the heuristic in Section 5. As an indicator Table 4 in Section 5 shows the number of times (in %) that the BMSTP-H algorithm reached the best lower bound for each BMSTP formulation.

3. Primal-dual BMSTP formulations

Let $\min_{x\geq 0}\{cx : x \in \mathcal{T}\}$ be a continuous linear MSTP formulation and $\max_{\mu\geq 0}\{d\mu : \mu \in \mathcal{T}^D(c)\}$ its dual form. We also denote by $\mathcal{T}^D(c,T)$ the polyhedron resulting by replacing c_e by T_e for each $e \in B$ of $\mathcal{T}^D$. We therefore can express the BMSTP as

$$F^0: \quad \max_{T\geq 0} \sum_{e\in B} T_e x_e \tag{3a}$$

$$\text{s.t.:} \quad x \in \mathcal{T} \tag{3b}$$

$$\mu \in \mathcal{T}^D(c,T) \tag{3c}$$

$$\sum_{e\in B} T_e x_e + \sum_{e\in R} c_e x_e = d\mu. \tag{3d}$$

Since $\mathcal{T}$ and $\mathcal{T}^D(c,T)$ define linear domains, the strong duality theorem holds and the primal and dual solutions can be forced to take the same value as in (3d). Therefore, the revenue (3a) can be optimized ensuring that an optimal minimum spanning tree is chosen by the follower. In addition, if c_e are integers, at least one optimal solution has integers variables T_e as well (as previously mentioned) and, therefore, integrality constraints are not needed for variables x.

Note that the product of T and x variables makes F^0 non-linear. In this section, we firstly propose two linearizations of F^0. Second we develop a polyhedral description of the Spanning Trees (ST) of G (coming from the one given by Martin, 1991) and its dual form. Finally we propose different alternatives to describe the spanning tree polyhedron $\mathcal{T}$ within F^0.

3.1. Linear BMSTP formulations

The product of variables Tx can be linearized defining a new set of variables $p_e = x_e T_e,\ e \in B$ as the profit of edge e. In the following we denote by F_p the linearization of (3a)–(3d) using variables p, that is

$$
\begin{aligned}
F_p:\ & \max_{T \geq 0} \sum_{e \in B} p_e && && \text{(4a)}\\
s.t.\ & x \in \mathcal{T} && && \text{(4b)}\\
& \mu \in \mathcal{T}^D(c, T) && && \text{(4c)}\\
& \sum_{e \in B} p_e + \sum_{e \in R} c_e x_e = d\mu. && && \text{(4d)}\\
& m_e x_e \leq p_e \leq M_e x_e && e \in B && \text{(4e)}\\
& p_e \leq T_e && e \in B && \text{(4f)}\\
& T_e \leq p_e + M_e(1 - x_e) && e \in B && \text{(4g)}\\
& m_e \leq T_e \leq M_e && e \in B && \text{(4h)}\\
& x_e \in \{0, 1\} && e \in E && \text{(4i)}
\end{aligned}
$$

Constraints (4e)–(4g) provide a linearization of the bilinear terms $p_e = x_e T_e,\ e \in B$ by means of the standard McCormick linearization (McCormick, 1976).

Note that once Tx is linearized, the integrality property of the problem is lost and integrality conditions (4i) are required.

We observe that variables T_e and p_e can be discretized, thus obtaining alternative formulations. Indeed, let $\{0, c^2, ..., c^{|K|}\}$ be the set made up of the zero value and the $|K - 1|$ different edge costs that appear in the initial red tree, where $k \in K$ is the index of the $k - th$ cost and K is a sorted set in non-decreasing order. More precisely, we can also define the set K for each edge as $K_e = \{k \in K : m_e \leq c^k \leq M_e\}$. Let z_e^k be a binary variable equal to one if and only if the price of edge e is equal to the k-th cost, that is,

$$T_e = \sum_{k \in K_e} z_e^k c^k \ \ e \in B. \tag{5}$$

Analogously, the values of p can be also discretized by using the binary variable $\bar{z}_e^k$ equal to one if and only if edge e is priced with the $k - th$ cost, that is,

$$p_e = \sum_{k \in K_e} \bar{z}_e^k c^k,\ \ e \in B. \tag{6}$$

Therefore F_p can be rewritten as follows:

$$
\begin{aligned}
F_z:\ & \max \sum_{e\in B}\sum_{k\in K_e} \bar{z}_e^k c^k && && \text{(7a)}\\
& s.t.\ \ x\in\mathcal{T} && && \text{(7b)}\\
& \mu\in\mathcal{T}^D(c,z) && && \text{(7c)}\\
& \sum_{e\in B}\sum_{k\in K_e}\bar{z}_e^k c^k + \sum_{e\in R} c_e x_e = d\mu && && \text{(7d)}\\
& \sum_{k\in K_e} z_e^k = 1 && e\in B && \text{(7e)}\\
& \sum_{k\in K_e} \bar{z}_e^k = x_e, && e\in B && \text{(7f)}\\
& \bar{z}_e^k \le z_e^k, && e\in B, k\in K_e && \text{(7g)}\\
& z_e^k \le \bar{z}_e^k + (1-x_e), && e\in B, k\in K_e && \text{(7h)}\\
& z_e^k, \bar{z}_e^k \in\{0,1\} && e\in E, k\in K_e && \text{(7i)}\\
& x_e\in\{0,1\} && e\in E && \text{(7j)}
\end{aligned}
$$

Constraint (7e) ensures that each edge is priced with one of the costs. Constraint (7f) ensures that each edge leaves a profit coming from one of the costs if the edge is chosen by the follower and a null profit otherwise. Constraint (7g) ensures that the benefit of a blue edge is lower or equal than the profit of the edge. Constraint (7h) together with (7g) ensures that if a blue edge is chosen, its profit is equal to its price.

In addition, we note in constraints (7c) by $\mathcal{T}^D(c,z)$ the polyhedron coming from $\mathcal{T}^D(c,T)$ when variables T_e, $e\in B$ are replaced by the values given in (5).

Therefore, note that (4e)–(4h) are implied by (7e)-(7j) as we show in the following property.

Property 6. *Let $\Omega_{LR}^{p,T;x}$ be the projection of the polyhedron defined by constraints* (4b)–(4g) *over the x variables and $\Omega_{LR}^{z,\bar{z};x}$ the projection of the polyhedron given by* (7b)–(7h) *over the x variables. Then $\Omega_{LR}^{z,\bar{z};x}\subseteq\Omega_{LR}^{p,T;x}$.*

Proof.

$\subseteq$) Let $(x,z,\bar{z})\in\Omega_{LR}^{z,\bar{z};x}$, we prove that $(x,z,\bar{z})$ verifies (4e)–(4h).

First, by replacing relations (5) *and* (6) *in* (7c) *and* (7d) *we obtain* (4c) *and* (4d).

Now, we prove that $(x,z,\bar{z})$ verifies (4e), (4f) *and* (4g)*:*

- *For each $e\in B$,* (6) $\Rightarrow p_e=\sum_{k\in K_e} c^k\bar{z}_e^k \le M_e\sum_{k\in K_e}\bar{z}_e^k \overset{(7e)}{=} M_e x_e \Rightarrow$ (4e).
- *For each $e\in B, k\in K_e$,* (7g) $\Rightarrow \bar{z}_e^k\le z_e^k \Rightarrow c^k\bar{z}_e^k\le c^k z_e^k \Rightarrow \sum_{k\in K_e} c^k\bar{z}_e^k \le \sum_{k\in K_e} c^k z_e^k \Rightarrow p_e\le T_e\Rightarrow$ (4f).
- *For each $e\in B, k\in K_e$,* (7g) $\Rightarrow 0\le z_e^k-\bar{z}_e^k\le\sum_{k\in K_e} c^k(z_e^k-\bar{z}_e^k)\le M_e\sum_{k\in K_e}(z_e^k-\bar{z}_e^k) = M_e(1-x_e)\Rightarrow$ (4g).

$\square$

3.2. KM-MST formulation and dual form

We recall that the general schemes proposed in the previous section require a linear STP formulation, $\mathcal{T}$, and its dual form, $\mathcal{T}^D$. For that aim, we present next the Kipp Martim MSTP formulation and its dual form.

Martin (1991) proposes a TDI formulation for the MST problem with a number of variables and constraints which are polynomial in the input size. For this, an arborescence rooted at each vertex $k \in V$ is modeled. The arcs of such arborescences are then related to the design variables x defined above. For $k \in V, (u,v) \in E$, let us denote by q_{kuv} and q_{kvu} to the decision variables that respectively indicate whether or not arc (u,v) and (v,u) belong to the arborescence rooted at k, where A stands for the set or arcs. Then, the formulation is as follows:

$$\min \sum_{e \in E} c_e x_e \tag{8a}$$

$$s.t. \quad \sum_{e \in E} x_e = n-1 \tag{8b}$$

$$\sum_{s \in V:(k,s) \in A} q_{kks} \leq 0 \qquad k \in V \tag{8c}$$

$$\sum_{v \in V:(u,v) \in A} q_{kuv} \leq 1 \qquad k,u \in V : u \neq k \tag{8d}$$

$$q_{kuv} + q_{kvu} = x_{uv} \qquad k \in V, (u,v) \in E \tag{8e}$$

$$x_{uv} \geq 0 \qquad (u,v) \in E \tag{8f}$$

$$q_{kuv} \geq 0 \qquad k \in V, (u,v) \in A \tag{8g}$$

Constraint (8a) ensures that the tree has $n-1$ edges. Constraints (8b)-(8d) break cycles that could be generated by the $n-1$ edges. Note that if there is a cycle of undirected edges containing vertex k, then by (8b) there is a corresponding cycle of directed edges defined by q_{kij} which also contains vertex k (refer to this set of directed arcs as the k-arcs). However, there cannot be a cycle of k-arcs which contains vertex k. This is impossible by (8d) and the cycle cannot be directed. If the cycle is not directed, then there is at least one vertex i with two k-arcs directed out of it. This is impossible by (8c). Thus, there are no cycles in the solution and we have by (8a) a spanning tree.

Formulation (8a)-(8f) can be simplified removing variables $q_{kkv} = 0, k \in V, (k,v) \in A$ as follows:

$$
\begin{aligned}
&\min \sum_{e\in E} c_e x_e && && (9a)\\
&\quad \sum_{e\in E} x_e = n-1 && && (9b)\\
&\quad \sum_{\substack{(u',v)\in E: (u'=k\wedge v=u)\vee\\ (u'=u\wedge v=k)}} x_{u'v} + \sum_{(u,v)\in A: v\neq k} q_{kuv} \leq 1 && k,u\in V: u\neq k && (9c)\\
&\quad q_{kuv} + q_{kvu} = x_{uv} && k\in V, (u,v)\in E: u,v\neq k && (9d)\\
&\quad x_{uv} \geq 0 && (u,v)\in E && (9e)\\
&\quad q_{kuv} \geq 0 && k\in V, (u,v)\in A: v\neq k && (9f)
\end{aligned}
$$

In the following, we denote by $\mathcal{T}^{km}$ to the polyhedron associated to the KM-MSTP formulation, that is, equations (9b)–(9f). In addition, the dual of (9a)–(9f) can be written as:

$$
\begin{aligned}
&\max \alpha(n-1) - \sum_{k\in V} \sum_{u\in V: v\neq k} \beta_{ku} && && (10a)\\
&\quad \alpha - \beta_{uv} - \beta_{vu} - \sum_{k'\in V: k'\neq u,v} \gamma_{uv}^{k'} \leq c_{uv} && (u,v)\in E && (10b)\\
&\quad -\beta_{ku} + \sum_{\substack{(u',v')\in E: (u'=u\wedge v'=v)\vee\\ (u'=v\wedge v'=u)}} \gamma_{u'v'}^{k} \leq 0 && k\in V, (u,v)\in A: u,v\neq k && (10c)\\
&\quad \beta_{ku} \geq 0 && k,u\in V: u\neq k && (10d)\\
&\quad \gamma_{uv}^{k} && k\in V, (u,v)\in E: u,v\neq k && (10e)
\end{aligned}
$$

Finally, we denote by $\mathcal{T}^{D}$ to the polyhedron associated to the KM-MSTP dual formulation, that is, equations (10b)–(10e). Note that now, both $\mathcal{T}^{km}$ and $\mathcal{T}^{D}$ can be implemented in previous BMSTP formulations to set effective valid representations.

3.3. Other primal-dual BMSTP formulations

Previous BMSTP formulations assume constraint $x \in \mathcal{T}^{km}$. Note that this constraint can be replaced by any other STP polyhedron, namely *subtour elimination*, *Miller-Tucker-Zemlim* (MTZ), *flow*, etc (see Magnanti and Wolsey, 1995; Fernández et al., 2017). For example, the Miller-Tucker-Zemlim formulation (Miller et al., 1960) uses variables y_{uv} which take the value 1 if and only if arc (u,v) belong to the arborescence and continuous variables l_u, denoting the position that node u occupies in the arborescence with respect to the root node.

It is well-known that STP formulations exist with the integrality property. Unfortunately, when they are embedded within the BMSTP framework the integrality property is lost. Then, alternative STP formulations without such property may now be superior (in computational terms) and this explains why some of the formulations we have used lack the integrality property. In Table 0 we resume the main properties of the STP formulations that we have considered. The criteria that have guided the selection of the formulations are either their good theoretical properties or some characteristic that seemed useful as, for instance, a small number of variables or constraints.

Table 0
Main properties of the MST formulations considered

Formulation	notation	main constraints	root	# vars	# const.	int
Subtour Edmonds (1970)	$\mathcal{T}^{sub}$	$\sum_{e\in E(S)} x_e \leq \lvert S\rvert - 1,\ \emptyset \neq S \subset V$		$O(\lvert E\rvert)$	$Exp(n)$	Yes
Kipp Martin Martin (1991)	$\mathcal{T}^{km}$	$\sum_{(u,v)\in\delta^+(u)} q_{kuv} \leq \begin{cases} 1, & k \in V, u \in V : u \neq k \\ 0, & k \in V, u = k \end{cases}$	$\forall k$	$O(n\lvert E\rvert)$	$O(n\lvert E\rvert)$	Yes
Miller-Tucker-Zemlim Miller et al. (1960)	$\mathcal{T}^{mtz}$	$l_v \geq l_u + 1 - n(1 - y_{uv}),\ (u,v) \in A$	r	$O(\lvert E\rvert)$	$O(\lvert E\rvert)$	No
Flow Gavish (1983)	$\mathcal{T}^{flow}$	$\sum_{(u,v)\in\delta^+(u)} \varphi_{uv} - \sum_{(v,u)\in\delta^-(u)} \varphi_{vu} = \begin{cases} n-1, & u = r \\ -1, & u \in V \setminus \{r\} \end{cases}$	r	$O(\lvert E\rvert)$	$O(\lvert E\rvert)$	No
KM extended Fernández et al. (2017)	$\mathcal{T}^{km2}$	$\sum_{(u,v)\in\delta^+(u)} q_{uv} \leq \begin{cases} 1, & u \in V : u \neq r \\ 0, & u = r \end{cases}$	r	$O(\lvert E\rvert)$	$Exp(n)$	Yes

In particular, the subtour and Kipp Martin formulations present the advantage of the integrality property (as you see in the last column) but they exhibit the inconvenience of an exponential number of constraints in the case of the subtour, or a cubic number of variables and constraints as it is the case of Kipp Martin. As an alternative, the MTZ and FLow formulations present lower dimensions and are, therefore, easier to handle.

We propose as an alternative STP formulation a relaxation of Kipp Martin that instead of building an arborescence at each node, it builds only one of them (see Fernández et al., 2017). Therefore cut-set inequalities will be required to be included dynamically in a Branch&Cut algorithm. The separation of this inequalities can be carried out in polynomial time by finding the cut Gomory-Hu tree.

4. A path-based BMSTP formulation

In this section we present an alternative BMSTP formulation that does not require the strong duality property. Instead, we impose minimum cost optimality constraints to a path-based STP formulation. In this way, we can impose in the objective function a maximization of the benefits of the blue edges which in turns ensures the validity of this approach.

First, we propose the announced path-based formulation for the STP. Let P denote the set of pairs of nodes such that $i < j$. We define now φ_{uv}^{ij} as the flow through edge (u, v) going from origin i to destination j with $(i,j) \in P$. The following set of constraints define a polyhedral description of the spanning trees of G.

$$\mathcal{T}^{path}: \sum_{v \in V:(i,v)\in A} \varphi_{iv}^{ij} = 1 \qquad (i,j) \in P \tag{11a}$$

$$\sum_{(u,v)\in A} \varphi_{uv}^{ij} - \sum_{(v,u)\in A} \varphi_{vu}^{ij} = 0 \qquad (i,j) \in P, v \in V : v \neq i,j \tag{11b}$$

$$\sum_{(u,j)\in A} \varphi_{uj}^{ij} = 1 \qquad (i,j) \in P \tag{11c}$$

$$\varphi_{uv}^{ij} + \varphi_{vu}^{ij'} \leq x_{uv} \qquad (i,j) \in P, (i,j') \in P, (u,v) \in E : u,v \neq i,j \tag{11d}$$

$$\sum_{(u,v)\in E} x_{uv} = n-1 \tag{11e}$$

$$\varphi_{uv}^{ij} \geq 0 \qquad (i,j) \in P, (u,v) \in A : v \neq i, u \neq j \tag{11f}$$

$$0 \leq x_e \leq 1 \qquad e \in E \tag{11g}$$

Formulation (11a)-(11g) defines a tree on the graph G. Constraints (11a)-(11c) guarantee that the flow subnetwork is connected. Constraints (11d) ensure that an edge is selected if there is a positive flow traversing any of its arcs.

Note that we do not define variables φ_{ui}^{ij} and φ_{jv}^{ij} in $\mathcal{T}^{path}$ since flow that is sent from i to j does not arrive to i or depart from j. In addition, constraints (11d) can alternatively be defined as

$$\varphi_{uv}^{ij} + \varphi_{vu}^{ij} \leq x_{uv} \ (i,j) \in P, (u,v) \in E : u,v \neq i,j. \tag{12}$$

Replacing (11d) by (12) reduces significantly the number of variables and constraints in $\mathcal{T}^{path}$. However, with this enhancement the integrality property of variables x in $\mathcal{T}^{path}$ is lost.

Next, we can add an additional constraint to $\mathcal{T}^{path}$ in order to guarantee this tree to be optimal (minimal cost):

$$(\varphi_{uv}^{ij} + \varphi_{vu}^{ij})c_{uv} \leq c_{ij}(1 - x_{ij}) \qquad (i,j) \in P, (u,v) \in E : (u,v) \neq (i,j) \tag{13a}$$

Constraint (13a) impose that if there is flow sent from i to j along arcs (u,v) or (v,u) (that is $\varphi_{uv}^{ij} + \varphi_{vu}^{ij} \neq 0$, what implies $x_{ij} = 0$) then $c_{uv} \leq c_{ij}$. This let us formulate the BMSTP as follows:

$$
\begin{aligned}
F^{path}: \quad & \max \sum_{e \in B} T_e x_e && && \text{(14a)}\\
s.t. \quad & (x, \varphi) \in \mathcal{T}^{path} && && \text{(14b)}\\
& (\varphi_{uv}^{ij} + \varphi_{vu}^{ij})(c_{uv} + T_{uv}) \leq (T_{ij} + c_{ij})(1 - x_{ij}) && (i,j) \in E, (u,v) \in E : (u,v) \neq (i,j) && \text{(14c)}\\
\text{vars:} \quad & x_e \in \{0, 1\} && e \in E && \text{(14d)}\\
& \varphi_{uv}^{ij} \geq 0 && (i,j) \in P, (u,v) \in A && \text{(14e)}\\
& T_e \geq 0 && e \in B && \text{(14f)}
\end{aligned}
$$

Note that for each $(u, v) \in E$ constraints (14c) are well defined because we assume $c_e = 0, e \in B$ and T_e only defined in B.

The linear formulation that removes the non-linearity of the product of p and T variables and the product of φ and T variables can be obtained by standard techniques. However, adding the new required inequalities destroys the integrality property of the tree polytope. Therefore, to have a valid formulation, constraint (15m) must be augmented.

$$
\begin{aligned}
F_p^{path}: \quad & \max \sum_{e \in B} p_e && && \text{(15a)}\\
s.t. \quad & (x,\varphi) \in \mathcal{T}^{path} && && \text{(15b)}\\
& p_e \le M_e x_e && e \in B && \text{(15c)}\\
& p_e \le T_e && e \in B && \text{(15d)}\\
& T_e \le p_e + M_e(1 - x_e) && e \in B && \text{(15e)}\\
& t_{uv}^{ij} \le (\varphi_{uv}^{ij} + \varphi_{uv}^{ij}) M_e && (i,j) \in P, (u,v) \in B && \text{(15f)}\\
& t_{uv}^{ij} \le T_{uv} && (i,j) \in P, (u,v) \in B && \text{(15g)}\\
& T_{uv} \le t_{uv}^{ij} + M_e(1 - \varphi_{uv}^{ij} - \varphi_{vu}^{ij}) && (i,j) \in P, (u,v) \in B && \text{(15h)}\\
& (\varphi_{uv}^{ij} + \varphi_{vu}^{ij}) c_{uv} \le c_{ij}(1 - x_{ij}) && (i,j) \in R, (u,v) \in R : (u,v) \neq (i,j) && \text{(15i)}\\
& t_{uv}^{ij} \le c_{ij}(1 - x_{ij}) && (i,j) \in R, (u,v) \in B : (u,v) \neq (i,j) && \text{(15j)}\\
& (\varphi_{uv}^{ij} + \varphi_{vu}^{ij}) c_{uv} \le T_{ij} - p_{ij} && (i,j) \in B, (u,v) \in R : (u,v) \neq (i,j) && \text{(15k)}\\
& t_{uv}^{ij} \le T_{ij} - p_{ij} && (i,j) \in B, (u,v) \in B : (u,v) \neq (i,j) && \text{(15l)}\\
& x_e \in \{0,1\} && e \in E && \text{(15m)}\\
& \varphi_{uv}^{ij} \ge 0 && (i,j) \in P, (u,v) \in A && \text{(15n)}\\
& T_e \ge 0 && e \in B && \text{(15o)}\\
& t_{uv}^{ij} \ge 0 && (i,j) \in P, (u,v) \in A && \text{(15p)}\\
& p_e \ge 0 && e \in B && \text{(15q)}
\end{aligned}
$$

Constraints (15c)–(15e) provide a linearization of the product $p_e = x_e T_e$, $e \in B$ by means of the standard McCormick linearization. Constraints (15f)–(15h) provide a linearization of the product φT by means of variable t defined as

$$t_{uv}^{ij} = (\varphi_{uv}^{ij} + \varphi_{vu}^{ij}) T_{uv} \ \ (i,j) \in P, (u,v) \in B \tag{16}$$

and by means of the standard McCormick linearization. Constraints (15i)–(15l) provide a linearization of (14c) by means of variable t and distinguishing the different cases where $(i,j) \in R$ or B and $(u,v) \in R$ or B.

There exists a similar formulation translating variables T and p into variables z and $\bar{z}$. Given that such formulation has not provided good results in computational terms, we skip its description for the sake of readability.

5. Computational results

Next, we report on the results of some computational experiments that we have run, in order to compare empirically the proposed formulations. We have studied the BMSTP combining the different formulations proposed for the STP.

Instances $G = (V, E)$ are generated as in Morais et al. (2016). We first generate a complete graph $G = (V, E^c)$ according to different values of $|V|$ and the components of the cost vectors are randomly chosen from a set K of $|K|$ random integers drawn from a uniform distribution on $[1, c_{max}]$. We then compute a MST solution $(V, \hat{E})$ and we initialize $E \leftarrow \hat{E}$, $R \leftarrow \hat{E}$, $B \leftarrow \emptyset$. Additional edges are then randomly picked from $E_c \backslash \hat{E}$, until a desired graph density d is obtained. If a given edge $e \in E_c \backslash \hat{E}$ is added to E, we choose with a probability p if $R \leftarrow R \cup \{e\}$ otherwise, $B \leftarrow B \cup \{e\}$. If e is added to R, the cost assigned to e (c_e) is randomly chosen from a set K of $|K|$ random integers. In particular, we choose $|V| \in \{20, 30, 50, 70\}$, $c_{max} = 150$, $d \in \{10\%, 20\%, 30\%, 50\%\}$, $p = 0.5$, and $|K| = \{3, 5, 7\}$. Note that in this case the mean value of $|B|$ is $\frac{|V|(|V|-2)}{2}$.

In all tables, reported results correspond to groups of 10 instances with the same triplet $(|V|, d, |C|)$. We present average results (and some maximum values) for each group. This way, in total, we have a set of 240 benchmark instances. All instances were solved with the MIP Xpress 7.7 optimizer, under a Windows 10 environment in an Intel(R) Core(TM)i7 CPU 2.93 GHz processor and 16 GB RAM. Default values were initially used for all parameters of Xpress solver and a CPU time limit of 1800 seconds was set. We have also tested different combinations of parameters for the solver cut strategy and intensity of heuristics but, unless it is specified, the best results were obtained with the parameters of the solver set to the default values. An initial solution was given to the problem by the three modules of BMSTP-H described in Section 2.2. The separation of the cutset inequalities in formulation $\mathcal{T}^{km2}$ was implemented using a max-flow based algorithm (Gusfield, 1990).

Tables are grouped in blocks. The first block contains three columns with the values of the instances parameters. Then, we give a block of 7 columns for each tested formulation. The columns of each block are the following:

1. Columns $gR\overline{L}$ give the percentage relative gap, computed as $100\frac{obj_R - obj_{\overline{L}}}{obj_R}$, where obj_R denotes the optimal value of the linear relaxation at the root node and $obj_{\overline{L}}$ denotes the best known lower bound obtained in all our experiments.
2. Columns $gU\overline{L}$ give the percentage relative gap, computed as $100\frac{obj_U - obj_{\overline{L}}}{obj_U}$, where now obj_U denotes the upper bound at termination.
3. Columns $g\overline{U}L$ give the percentage relative gap, computed as $100\frac{obj_{\overline{U}} - obj_L}{obj_{\overline{U}}}$, where now $obj_{\overline{U}}$ denotes the best known upper bound obtained in all our experiments and obj_L denotes the lower bound at termination.
4. Columns gUL give the percentage relative gap, computed as $100\frac{obj_U - obj_L}{obj_U}$.
5. Columns gUL^* give the maximum of the gUL values among the 10 instances of the row.
6. Columns $|\#|$ indicate the number of instances in the group that could be solved to optimality within the CPU time limit.
7. Columns nod indicate the average number of nodes explored in the branch-and-bound tree.

Note that, while $gR\overline{L}$ and $gU\overline{L}$ provide quality measures of the upper bounds (at the root node and at termination, respectively), and $g\overline{U}L$ provides a quality measure of the lower bounds. In addition, gUL and gUL^* provide measures of both upper and lower bounds for average and worst case performance respectively. Entries with the symbol "-" indicates that the average/maximum gaps are 0, or in other words, that all instances were solved to optimality.

The caption just below each block gives the formulation the block refers to. Throughout the section, F_{mor} denotes the formulation with the best results reported in Morais et al. (2016) for the BMSTP. Otherwise, we denote by $F_p^{(.)}$ the combination of the BMSTP F_p formulation together with a spanning tree polyhedron $\mathcal{T}^{(.)}$ (idem with $F_z^{(.)}$). In this way, we report results of formulations F_p^{flow}, F_p^{km}, F_p^{mtz}, F_p^{km2} and F_z^{flow}, F_z^{km}, F_z^{mtz}, F_z^{km2} that have shown the best performance in preliminary tests. Note that we do not report results of the path formulations F_p^{path} since those results were clearly outperformed by $F_p^{(.)}$ and $F_z^{(.)}$ in preliminary tests. Then, we have summarized the results in five tables.

1. Table 1 shows results for $F_p^{(.)}$ formulations, namely F_p^{flow}, F_p^{km}, F_p^{mtz} and F_p^{km2}.
2. Table 2 shows results for $F_z^{(.)}$ formulations, namely F_z^{flow}, F_z^{km}, F_z^{mtz} and F_z^{km2}.
3. Table 3 shows the number of times (in %) that the BMSTP-H algorithm reached the best lower bound for each formulation.
4. Table 4 choses the best blocks so far, and extends the time limit of some rows of these blocks from 1800 seconds to 5 hours.
5. Table 5 displays a comparison between the results reported in Morais et al. (2016) and the results obtained with our best formulation, with the same set of instances.

In order to facilitate the comparison among tables, best results in each table are marked in bold. In this sense, Tables 1 and 2 are treated as a sigle one, so as to highlight the best values among the eight proposed formulations, namely F_p^{flow}, F_p^{km}, F_p^{mtz} , F_p^{km2}, F_z^{flow}, F_z^{km}, F_z^{mtz} and F_z^{km2}.

In Table 1, the results of block F_p^{flow} exhibit the worst values of $gR\overline{L}$ in the group. However, most of these instances have similar values in terms of $g\overline{U}L$ than the other formulations of the group. As colummn nod shows, the number or required nodes to reach the optimal values in instances $(20, 30, |C|)$ is the biggest in the group and consequently, gaps are bigger than in other blocks. On the contrary, the results of block F_p^{km} exhibit the best values of $gR\overline{L}$. Howerver, in many cases these gaps only improve slightly (or not improved at all) other $gR\overline{L}$ values of the group. In addition, gaps gUL and gUL^* remain far from the best values of the group. Note that according to the low average number of explored nodes in the B&B tree, in particular for sizes $(|V|, d) = (70, 20)$, solving the LP relaxation of the problem becomes quite hard so the corresponding gaps at termination remain quite large in comparison with other formulations. We recall that F_p^{km} uses the largest number of variables and constraints, which can be too high in larger graphs. Block F_p^{mtz} shows good average gaps $gR\overline{L}$ and $g\overline{U}L$, and it also provides some best values of the group for gaps gUL^* in the largest instances. Block F_p^{km2} shows a similar performance than F_p^{mtz} in gaps $gR\overline{L}$ and $g\overline{U}L$, and it also provides the best values of the group for gaps $g\overline{U}L$, gUL and gUL^* in instances $|V| < 70$. Since many of the constraints in this formulation are added on the fly within the B&B search tree, we can observe that consecuently, the number of explored nodes is the largest of the group. From this table, we conclude that the most promising formulation is F_p^{km2} for medium to large size instances.

In Table 2, the results of block F_z^{flow} exhibit the worst values of $g\overline{RL}$ in the group. However, most of these instances do not remain far, in terms of $g\overline{UL}$, from the other formulations of the group. As in the $F_p^{(.)}$ case, we observe that F_z^{flow} shows nthe worst gap values in general terms. The results of block F_z^{km} exhibit the best values of $g\overline{RL}$ and in some cases, these gaps are improved giving rise to best values of gaps gUL and gUL^* in medium size instances. As in F_p^{km} solving the LP relaxation in sizes $(|V|, d) = (70, 10)$ becomes quite hard giving rise to no more than three explored nodes in the B&B tree for these cases. Block F_p^{mtz} shows good averge gaps $g\overline{RL}$, $g\overline{UL}$ and gUL^* but it is outperformed in general by block F_z^{km2} that shows the best performace of this group. From this table, we conclude that the most promising formulation is F_z^{km2} for medium to small size instances.

Table 3 shows the number of times (in %) that the B&B search returned the same lower bound as the one given by the BMSTP-H algorithm. We observe that the percentages are smaller for the $F_p^{(.)}$ formulations compared to the $F_z^{(.)}$ formulations. This means that $F_z^{(.)}$ formulations are harder to tackle in order to find feasible solutions (lower bounds) because the number of binary variables is highly superior to the one of the $F_p^{(.)}$ formulations. Besides, we conclude from this table the BMSTP-H algorithm provides a reasonably good feasible initial lower bound that in many cases is no able to be outperformed by the solver after 1800 seconds of running time.

Table 4 shows a comparison of the best blocks so far, namely F_p^{km2} and F_z^{km2}, with a time limit of 1800 seconds and a time limit of 5 hours. In this case, we display results for only some rows of instances that correspond to the same combinations of $(|V|, d, |C|)$ that were studied in Morais et al. (2016). Obviously, mostly all gaps are improved when the time limit is extended to 5 hours but we observe a bigger improvement for F_z^{km2} in small-medium instances and a bigger improvement for F_p^{km2} in medium-big instances.

Table 5 shows a comparison between the results provided in Morais et al. (2016) and the results with our best formulations F_p^{km2} and F_z^{km2}. Note that in this case, we have used the same set of instances as in Morais et al. (2016) and only results for a single instance per row are provided. Block F_{mor} shows the best results reported by Morais et al. (2016) implemented in C++ and tested with a 2.4GHz Intel XEON E5645 machine, with 32 GB of RAM, under Linux Operating System. From this table, we observe first that both F_p^{km2} and F_z^{km2} are able to provide better lower bounds $objL$ and upper bounds $objU$ than those in F_{mor}. Consequently, gaps gUL are smaller in mostly all cases showing also that 11 out of 16 instances were able to be solved to optimality. In addition, running times (displayed in column t) show that 6 out of 16 instances were solved to optimality for F_p^{km2} in less than 10 minutes. In general terms, we conclude that blocks F_p^{km2} and F_z^{km2} outperform significantly block F_{mor}.

6. Conclusions

In this paper we have presented different mathematical programming formulations for the BMSTP based on the properties of the MSTP and the bilevel optimization paradigm. We have established theoretical and empirical comparisons between these new formulations that have shown to be effective for efficiently solving medium to big size instances. In addition we are able to outperform previous existing computational results in the literature.

Acknowledgements

The research of the first author has been partially supported by the Fonds de la Recherche Scientifique - FNRS under Grant(s) no PDR T0098.18. The second author was partially supported by the V Plan Propio de Investigación (Universidad de Sevilla). The second and third authors were partially supported by project MTM2016-74983-C02-01 (MINECO/FEDER). This support is gratefully acknowledged.

We also would like to acknowledge Vinicius Morais, Alexandre Salles da Cunha and Philippe Mahey for providing us their set of instances presented in Morais et al. (2016) that we have used in the present paper.

References

Bilò, D., Gualà, L., Leucci, S., Proietti, G., 2015. Specializations and generalizations of the stackelberg minimum spanning tree game. *Theoretical Computer Science* 562, 643–657.

Cardinal, J., Demaine, E., Fiorini, S., Joret, G., Langerman, S., Newman, I., Weimann, O., 2011. The stackelberg minimum spanning tree game. *Algorithmica* 59(2), 129–144.

Cardinal, J., Demaine, E., Fiorini, S., Joret, G., Newman, I., Weimann, O., 2013. The stackelberg minimum spanning tree game on planar and bounded treewidth graphs. *Journal of Combinatorial Optimization* 25(1), 19–46.

Edmonds, J., 1970. Submodular functions, matroids, and certain polyhedra. In *Combinatorial Structures and their Applications*. Gordon and Breach, New York, pp. 69–87.

Fernández, E., Pozo, M.A., Puerto, J., Scozzari, A., 2017. Ordered weighted average optimization in multiobjective spanning tree problems. *European Journal Operational Research* 260, 3, 886 – 903.

Gassner, E., 2002. Maximal spannende Baumprobleme mit einer Hierarchie von zwei Entscheidungstrgern. Diploma thesis, Graz University of Technology.

Gavish, B., 1983. Formulations and algorithms for the capacitated minimal directed tree problem. *Journal of the ACM* 30, 118–132.

Gusfield, D., 1990. Very simple methods for all pairs network flow analysis. *SIAM Journal on Computing* 19 (1), 143–155.

van Hoesel, S., 2008. An overview of stackelberg pricing in networks. *European Journal of Operational Research* 189(3), 1393–1402.

Labbé, M., Marcotte, P., Savard, G., 1998. A bilevel model of taxation and its application to optimal highway pricing. *Management Science* 44(12), 608–622.

Labbé, M., Violin, A., 2013. Bilevel programming and price setting problems. *4OR* 11, 1, 1–30.

Magnanti, T.L., Wolsey, L.A., 1995. Optimal trees. *Handbooks in operations research and management science* 7, 503–615.

Martin, R., 1991. Using separation algorithms to generate mixed integer model reformulations. *Operations Research Letters* 10, 3, 119–128.

McCormick, G.P., 1976. Computability of global solutions to factorable nonconvex programs: Part i - convex underestimating problems. *Mathematical Programming* 10, 147–175.

Miller, C.E., Tucker, A.W., Zemlin, R.A., 1960. Integer programming formulation of traveling salesman problems. *Journal of the ACM* 7, 4, 326–329.

Morais, V., da Cunha, A.S., Mahey, P., 2016. A branch-and-cut-and-price algorithm for the stackelberg minimum spanning tree game. *Electronic Notes in Discrete Mathematics* 52, 309 – 316. {INOC} 2015 – 7th International Network Optimization Conference.

Roch, S., Savard, G., Marcotte, P., 2005. An approximation algorithm for stackelberg network pricing. *Networks* 46(1), 57–67.

von Stackelberg, H., 1934. *Marktform und Gleichgewicht (Market and Equilibrium)*. Springer, Vienna.

Table 1
BMSTP results for the $F_p^{(\cdot)}$ formulations.

$\|V\|$	d	$\|C\|$	$g\overline{RL}$	$g\overline{UL}$	$\overline{gU}L$	gUL	gUL^*	$\|\#\|$	nod
20	30	3	13.5	-	-	-	-	**10**	1e3
20	30	5	18.2	-	-	-	-	**10**	**4e4**
20	30	7	17.4	-	-	-	-	**10**	1e4
20	50	3	4.8	1.6	-	1.6	8.8	7	**2e5**
20	50	5	**7.1**	5.8	-	5.8	12	0	4e5
20	50	7	**8.9**	7.8	1.7	7.9	16	1	3e5
30	30	3	6.9	3.4	**0.3**	3.4	14.5	6	1e4
30	30	5	9.8	6	3.1	6.1	14.1	2	3e4
30	30	7	13.4	9.4	**6.8**	9.5	16.2	**0**	6e4
30	50	3	0.4	0.2	-	0.2	1.6	9	3e3
30	50	5	4.1	3.8	0.8	4.1	10.3	1	3e4
30	50	7	5.9	5.7	4.4	6.5	21	**0**	3e4
50	10	3	16.2	4.1	1	4.4	13	2	1e4
50	10	5	19.3	5.3	1.5	6.3	9.7	1	1e4
50	10	7	22.5	6.3	2.8	7.4	18.2	2	1e4
50	20	3	4.1	2.6	**0.6**	2.6	**6.6**	1	5e3
50	20	5	9.4	**8.2**	14.4	14.8	25.1	**0**	5e3
50	20	7	13.8	**12.1**	20.1	20.4	30.8	**0**	5e3
70	10	3	16.3	7.6	7.3	11.2	21.3	0	1e3
70	10	5	18.7	13.1	15.7	16.4	26.6	**0**	9e2
70	10	7	21	14.8	22.7	23.4	33.2	**0**	9e2
70	20	3	**1.1**	1.1	4.7	5.1	18.6	0	1e3
70	20	5	4.7	**4.6**	7.7	7.7	23.9	**0**	2e2
70	20	7	8.1	**8**	11.9	11.9	23.7	**0**	2e2

F_p^{flow}

$\|V\|$	d	$\|C\|$	$g\overline{RL}$	$g\overline{UL}$	$\overline{gU}L$	gUL	gUL^*	$\|\#\|$	nod
20	30	3	**3.9**	-	-	-	-	**10**	76
20	30	5	**9.5**	-	-	-	-	**10**	8e3
20	30	7	**8.5**	-	-	-	-	**10**	5e3
20	50	3	**3.4**	1.4	-	1.4	8.6	8	5e4
20	50	5	**7.1**	5.2	-	5.2	10.6	1	2e5
20	50	7	**8.9**	7.9	**1.6**	7.9	16	1	2e5
30	30	3	**4.4**	3.3	**0.3**	3.3	14.8	5	1e4
30	30	5	**7.2**	6	**2.9**	6	14	2	2e4
30	30	7	**10.1**	9.4	**6.8**	9.4	15.2	**0**	2e4
30	50	3	**0.2**	0.2	-	0.2	1.6	9	1e3
30	50	5	**3.8**	3.8	**0.5**	3.8	7.4	1	1e4
30	50	7	**5.7**	5.7	4.3	6.3	19.5	**0**	1e4
50	10	3	**3.7**	1.9	**0.6**	1.9	10.9	7	2e3
50	10	5	**5.5**	2.4	**0.5**	2.5	7	**5**	3e3
50	10	7	**8**	3.7	1.9	4	14.9	**5**	2e3
50	20	3	**2.8**	2.4	**0.6**	2.4	**6.6**	2	1e3
50	20	5	**8.2**	**8.2**	9.7	10.2	17.3	**0**	1e3
50	20	7	**12.1**	**12.1**	15.7	16	25.9	**0**	1e3
70	10	3	**7.3**	6.8	**6**	9.2	19.2	**1**	5e2
70	10	5	**13**	**12.4**	15.8	16	26.7	**0**	1e2
70	10	7	**15.1**	14.4	22	22.3	33.2	**0**	1e2
70	20	3	**1.1**	1.1	5.7	6.1	24.6	0	25
70	20	5	**4.6**	**4.6**	7.7	7.8	24.3	**0**	3
70	20	7	**8**	**8**	11.9	11.9	23.7	**0**	0

F_p^{km}

$\|V\|$	d	$\|C\|$	$g\overline{RL}$	$g\overline{UL}$	$\overline{gU}L$	gUL	gUL^*	$\|\#\|$	nod
20	30	3	4.8	-	-	-	-	**10**	1e2
20	30	5	**9.5**	-	-	-	-	**10**	1e4
20	30	7	9.2	-	-	-	-	**10**	9e3
20	50	3	**3.4**	1.8	-	1.8	8.1	6	1e5
20	50	5	**7.1**	5.2	-	5.2	11.1	1	3e5
20	50	7	**8.9**	8	1.7	8	16	1	3e5
30	30	3	**4.4**	3.2	**0.3**	3.2	15	5	2e4
30	30	5	**7.2**	6	**2.9**	6	14	1	5e4
30	30	7	**10.1**	9.4	6.9	9.5	15.2	**0**	5e4
30	50	3	**0.2**	0.2	-	0.2	1.6	9	4e3
30	50	5	**3.8**	3.8	**0.5**	3.8	7.6	1	3e4
30	50	7	**5.7**	5.7	4.5	6.5	21	**0**	3e4
50	10	3	4.4	2	**0.6**	2	10.4	7	2e3
50	10	5	6.3	2.9	**0.5**	2.9	7.2	4	7e3
50	10	7	8.6	4.7	1.7	4.8	13.2	4	7e3
50	20	3	**2.8**	2.5	**0.6**	2.5	**6.6**	1	4e3
50	20	5	**8.2**	**8.2**	9.9	10.3	16.6	**0**	5e3
50	20	7	**12.1**	**12.1**	15.5	15.8	**24.1**	**0**	5e3
70	10	3	7.5	6.5	6.4	9.3	**19.1**	**1**	9e2
70	10	5	13.1	12.8	15.4	15.9	27.3	**0**	8e2
70	10	7	15.2	14.5	22	22.3	**30.4**	**0**	7e2
70	20	3	**1.1**	1.1	1	1.4	**3.8**	0	7e2
70	20	5	**4.6**	**4.6**	5.4	5.4	**13.2**	**0**	4e2
70	20	7	**8**	**8**	**10.1**	**10.1**	**21.8**	**0**	4e2

F_p^{mtz}

$\|V\|$	d	$\|C\|$	$g\overline{RL}$	$g\overline{UL}$	$\overline{gU}L$	gUL	gUL^*	$\|\#\|$	nod
20	30	3	4.9	-	-	-	-	**10**	1e2
20	30	5	**9.5**	-	-	-	-	**10**	2e4
20	30	7	9.2	-	-	-	-	**10**	1e4
20	50	3	**3.4**	0.8	-	0.8	7.8	9	1e5
20	50	5	**7.1**	5.2	-	5.2	11.4	1	4e5
20	50	7	**8.9**	8	1.7	8	16	1	4e5
30	30	3	**4.4**	2.8	**0.3**	2.8	13.3	6	2e4
30	30	5	**7.2**	6.4	**2.9**	6.4	14	1	6e4
30	30	7	**10.1**	9	**6.8**	**9**	**13.3**	**0**	7e4
30	50	3	**0.2**	0.2	-	0.2	1.6	9	4e3
30	50	5	**3.8**	3.8	**0.5**	3.8	7.4	1	5e4
30	50	7	**5.7**	5.7	**4.2**	**6.2**	**18.5**	**0**	6e4
50	10	3	4.8	1.8	**0.6**	1.8	9.6	6	5e3
50	10	5	6.8	**2.3**	0.6	**2.4**	7.3	**5**	1e4
50	10	7	8.6	**3.4**	**1.6**	**3.4**	**10.5**	**5**	9e3
50	20	3	3	2.7	**0.6**	2.7	**6.6**	0	8e3
50	20	5	**8.2**	**8.2**	**8.7**	**9.1**	**14.5**	**0**	8e3
50	20	7	**12.1**	**12.1**	**14.3**	**14.7**	26.6	**0**	9e3
70	10	3	7.5	6.7	**6**	**9.1**	19.5	**1**	1e3
70	10	5	13.1	12.5	**14.4**	**14.6**	**23.7**	**0**	9e2
70	10	7	15.2	14.6	**19.9**	**20.4**	32.7	**0**	1e3
70	20	3	**1.1**	1.1	**0.9**	**1.3**	**3.8**	0	5e2
70	20	5	**4.6**	**4.6**	**5.2**	**5.3**	13.8	**0**	6e2
70	20	7	**8**	**8**	10.8	10.8	**21.8**	**0**	8e2

F_p^{km2}

Table 2
BMSTP results for the $F_z^{(\cdot)}$ formulations.

$\|V\|$	d	$\|C\|$	$g\overline{RL}$	$g\overline{UL}$	$\overline{gU}L$	gUL	gUL^*	$\|\#\|$	nod
20	30	3	13.5	-	-	-	-	**10**	**1e3**
20	30	5	18.2	-	-	-	-	**10**	1e4
20	30	7	17.4	-	-	-	-	**10**	8e4
20	50	3	4.8	0.7	-	0.7	6.8	9	5e4
20	50	5	**7.1**	2.5	0.1	2.6	8	1	**5e5**
20	50	7	**8.9**	4.2	2.6	5.1	10.5	0	**5e5**
30	30	3	6.9	2.1	1.1	2.9	12	6	5e4
30	30	5	9.8	5	4.2	6.3	21.8	1	**8e4**
30	30	7	13.4	8.2	11.9	13.3	24.9	**0**	9e4
30	50	3	0.4	0.2	0.2	0.3	3.1	9	**1e4**
30	50	5	4.1	3.8	6.1	9.1	29.5	1	7e4
30	50	7	5.9	5.7	9.1	11	21	**0**	8e4
50	10	3	16.2	2.7	1.1	3.2	13.5	3	**1e4**
50	10	5	19.3	5.7	2.4	7.5	10.5	0	2e4
50	10	7	22.5	7.1	3.7	9	24.2	1	**1e4**
50	20	3	4.1	2.4	6.5	8	25.2	2	1e4
50	20	5	9.4	**8.2**	16.1	16.5	26.4	**0**	1e4
50	20	7	13.8	**12.1**	20.8	21.1	30.8	**0**	1e4
70	10	3	16.3	**5.8**	7.7	9.9	21.2	**1**	**1e3**
70	10	5	18.7	13.2	16.6	17.5	28.1	**0**	**1e3**
70	10	7	21	14.9	22.8	23.6	33.2	**0**	**1e3**
70	20	3	**1.1**	1.1	9.3	9.7	28.4	0	2e3
70	20	5	4.7	**4.6**	7.7	7.8	24.3	**0**	3e3
70	20	7	8.1	**8**	12.1	12.1	24.8	**0**	2e3

F_z^{flow}

$\|V\|$	d	$\|C\|$	$g\overline{RL}$	$g\overline{UL}$	$\overline{gU}L$	gUL	gUL^*	$\|\#\|$	nod
20	30	3	**3.9**	-	-	-	-	**10**	52
20	30	5	**9.5**	-	-	-	-	**10**	5e3
20	30	7	**8.5**	-	-	-	-	**10**	1e4
20	50	3	**3.4**	0.8	-	0.8	8.1	9	1e4
20	50	5	**7.1**	0.8	0.6	1.4	8.1	5	7e4
20	50	7	**8.9**	3.9	2	4.2	10.1	**3**	1e5
30	30	3	**4.4**	1	**0.3**	1	7.2	**8**	3e3
30	30	5	**7.2**	5.7	4	6.7	22.1	2	1e4
30	30	7	**10.1**	9.4	10.3	12.9	24.9	**0**	1e4
30	50	3	**0.2**	-	-	-	-	**10**	1e3
30	50	5	**3.8**	3.6	5.4	8.3	29.5	2	1e4
30	50	7	**5.7**	5.7	6.4	8.4	21	**0**	1e4
50	10	3	**3.7**	1.9	**0.6**	1.9	10	6	1e3
50	10	5	**5.5**	2.6	0.6	2.7	**6.6**	4	2e3
50	10	7	**8**	4.8	2	5.2	13.3	3	3e3
50	20	3	**2.8**	2.3	4	5.5	25.2	**3**	7e2
50	20	5	**8.2**	**8.2**	16	16.4	26.4	**0**	1e3
50	20	7	**12.1**	**12.1**	20.6	20.9	30.8	**0**	9e2
70	10	3	**7.3**	6.1	7.8	10.3	19.7	**1**	1e2
70	10	5	**13**	12.6	16.3	16.6	25.5	**0**	1e2
70	10	7	**15.1**	**14.2**	22.7	22.8	33.2	**0**	1e2
70	20	3	**1.1**	1.1	9.2	9.6	28.4	0	3
70	20	5	**4.6**	**4.6**	7.7	7.8	24.3	**0**	3
70	20	7	**8**	**8**	12.1	12.1	24.8	**0**	3

F_z^{km}

$\|V\|$	d	$\|C\|$	$g\overline{RL}$	$g\overline{UL}$	$\overline{gU}L$	gUL	gUL^*	$\|\#\|$	nod
20	30	3	4.8	-	-	-	-	**10**	2e2
20	30	5	**9.5**	-	-	-	-	**10**	7e3
20	30	7	9.2	-	-	-	-	**10**	2e4
20	50	3	**3.4**	0.7	-	0.7	6.7	9	3e4
20	50	5	**7.1**	0.7	-	0.7	6.2	**7**	2e5
20	50	7	**8.9**	**2.3**	1.7	**2.4**	5.8	**3**	2e5
30	30	3	**4.4**	1.1	**0.3**	1.1	5.7	7	**5e4**
30	30	5	**7.2**	4.7	3.5	5.3	14	2	4e4
30	30	7	**10.1**	8.5	9.1	10.8	19	**0**	6e4
30	50	3	**0.2**	-	-	-	-	**10**	6e2
30	50	5	**3.8**	2.4	2.3	4.1	18.5	**5**	2e4
30	50	7	**5.7**	5.7	7.2	9	21	**0**	6e4
50	10	3	4.4	**1**	0.8	**1.2**	12.2	**9**	1e3
50	10	5	6.3	2.5	1.1	3.1	12.8	3	9e3
50	10	7	8.6	5.2	2.4	5.9	13.5	3	8e3
50	20	3	**2.8**	2.4	2.3	4	13.9	2	4e3
50	20	5	**8.2**	**8.2**	15.1	15.6	26.4	**0**	8e3
50	20	7	**12.1**	**12.1**	20.7	21	30.8	**0**	8e3
70	10	3	7.5	6.4	7.8	10.5	19.7	**1**	6e2
70	10	5	13.1	12.5	16.5	16.6	26	**0**	6e2
70	10	7	15.2	14.7	22.8	23.3	33.2	**0**	7e2
70	20	3	**1.1**	1.1	4.3	4.6	28.4	0	1e3
70	20	5	**4.6**	**4.6**	7.7	7.8	24.3	**0**	2e3
70	20	7	**8**	**8**	12.1	12.1	24.8	**0**	1e3

F_z^{mtz}

$\|V\|$	d	$\|C\|$	$g\overline{RL}$	$g\overline{UL}$	$\overline{gU}L$	gUL	gUL^*	$\|\#\|$	nod
20	30	3	4.9	-	-	-	-	**10**	5e2
20	30	5	**9.5**	-	-	-	-	**10**	1e4
20	30	7	9.2	-	-	-	-	**10**	**1e5**
20	50	3	**3.4**	-	-	-	-	**10**	2e4
20	50	5	**7.1**	**0.2**	0.4	**0.6**	**4.2**	6	2e5
20	50	7	**8.9**	2.4	1.8	2.5	**5.5**	**3**	3e5
30	30	3	**4.4**	**0.7**	0.4	**0.8**	**4.8**	**8**	2e4
30	30	5	**7.2**	**3.6**	3.1	**3.8**	**11.8**	**3**	6e4
30	30	7	**10.1**	**7.5**	10.5	11.2	23.9	**0**	**1e5**
30	50	3	**0.2**	-	-	-	-	**10**	2e2
30	50	5	**3.8**	**1.5**	**0.5**	**1.6**	**4.6**	4	**8e4**
30	50	7	**5.7**	**4.7**	7.3	8.3	20	**0**	**1e5**
50	10	3	4.8	1.2	0.8	1.4	**8.6**	6	6e3
50	10	5	6.8	3	1.2	3.7	12.2	2	**2e4**
50	10	7	8.6	4.5	2.3	5.2	14	3	1e4
50	20	3	3	**1.6**	1.3	**2.3**	**6.6**	2	**2e4**
50	20	5	**8.2**	**8.2**	15.3	15.7	26.4	**0**	**1e4**
50	20	7	**12.1**	**12.1**	21	21.3	30.8	**0**	**1e4**
70	10	3	7.5	6.4	7.1	10	19.3	**1**	8e2
70	10	5	13.1	12.7	16.6	17	28.1	**0**	1e3
70	10	7	15.2	14.6	22.8	23.3	33.2	**0**	1e3
70	20	3	**1.1**	**0.9**	8.3	8.5	28.4	**2**	**2e3**
70	20	5	**4.6**	**4.6**	7.7	7.8	24.3	**0**	**5e3**
70	20	7	**8**	**8**	12.1	12.1	24.8	**0**	**4e3**

F_z^{km2}

Table 3
BMSTP-H results for $F_p^{(\cdot)}$ and $F_z^{(\cdot)}$ formulations: % of times BMSTP-H algorithm returned the lower bound of the B&B search

$\|V\|$ d	F_p^{flow}	F_p^{km}	F_p^{mtz}	F_p^{km2}	F_z^{flow}	F_z^{km}	F_z^{mtz}	F_z^{km2}
20 30	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3
20 50	23.3	23.3	23.3	23.3	26.7	26.7	26.7	26.7
30 30	30.0	30.0	33.3	26.7	66.7	50.0	50.0	33.3
30 50	33.3	30.0	33.3	30.0	73.3	60.0	50.0	43.3
50 10	30.0	20.0	16.7	16.7	46.7	16.7	20.0	30.0
50 20	43.3	16.7	20.0	16.7	80.0	70.0	70.0	70.0
70 10	60.0	50.0	50.0	43.3	93.3	86.7	90.0	90.0
70 20	63.3	66.7	43.3	50.0	100.0	96.7	76.7	83.3
total	39.6	33.8	31.7	30.0	65.0	55.0	52.1	51.3

Table 4
BMSTP results comparison for the best formulations with time limits of 0.5h and 5h.

$\|V\|$ d $\|C\|$	$gR\overline{L}$	$gU\overline{L}$	$g\overline{U}L$	gUL	gUL^*	$\|\#\|$	nod	$gR\overline{L}$	$gU\overline{L}$	$g\overline{U}L$	gUL	gUL^*	$\|\#\|$	nod	$gR\overline{L}$	$gU\overline{L}$	$g\overline{U}L$	gUL	gUL^*	$\|\#\|$	nod	$gR\overline{L}$	$gU\overline{L}$	$g\overline{U}L$	gUL	gUL^*	$\|\#\|$	nod
20 30 7	**9.2**	-	-	-	-	**10**	1e4	**9.2**	-	-	-	-	**10**	**1e5**	**9.2**	-	-	-	-	**10**	1e4	**9.2**	-	-	-	-	**10**	**1e5**
20 50 3	**3.4**	0.8	-	0.8	7.8	9	1e5	**3.4**	-	-	-	-	**10**	2e4	**3.4**	0.3	-	0.3	3.4	9	**3e5**	**3.4**	-	-	-	-	**10**	2e4
20 50 5	**7.1**	5.2	-	5.2	11.4	1	4e5	**7.1**	0.2	0.4	0.6	4.2	6	2e5	**7.1**	4.2	-	4.2	10.2	3	**3e6**	**7.1**	-	-	-	**0.2**	**9**	3e6
30 30 3	**4.4**	2.8	**0.3**	2.8	13.3	6	2e4	**4.4**	0.7	0.4	0.8	4.8	8	2e4	**4.4**	2.2	**0.3**	2.2	9.7	6	**2e5**	**4.4**	**0.3**	**0.3**	**0.3**	**2.8**	**9**	1e5
30 50 3	**0.2**	0.2	-	0.2	1.6	9	4e3	**0.2**	-	-	-	-	**10**	2e2	**0.2**	-	-	-	-	**10**	**1e4**	**0.2**	-	-	-	-	**10**	2e2
30 50 5	**3.8**	3.8	**0.6**	3.8	7.4	1	5e4	**3.8**	1.5	0.7	1.6	4.6	4	8e4	**3.8**	3.5	**0.6**	3.5	7.4	2	5e5	**3.8**	**0.6**	0.7	**0.7**	**3.4**	**7**	**7e5**
30 50 7	**5.7**	5.7	4.2	6.2	18.5	**0**	6e4	**5.7**	4.7	7.3	8.3	20	**0**	1e5	**5.7**	5.7	**3.7**	**5.7**	**16.1**	**0**	6e5	**5.7**	**3.7**	6.2	6.2	16.8	**0**	**1e6**
50 10 5	**6.8**	2.3	1.1	2.4	7.3	5	1e4	**6.8**	3	1.7	3.7	12.2	2	2e4	**6.8**	**1.3**	**1**	**1.3**	**5.3**	**7**	6e4	**6.8**	2.1	1.6	2.7	12.2	6	**1e5**
50 10 7	**8.6**	3.4	**2.2**	3.4	10.5	5	9e3	**8.6**	4.5	2.9	5.2	14	3	1e4	**8.6**	**2.2**	**2.2**	**2.2**	**9**	**6**	8e4	**8.6**	3.2	2.3	3.3	11	4	**1e5**
50 20 3	**3**	2.7	**0.8**	2.7	6.6	0	8e3	**3**	1.6	1.5	2.3	6.6	2	2e4	**3**	2.5	**0.8**	2.5	6.6	2	6e4	**3**	**0.9**	1	**1.1**	**3.4**	**4**	**2e5**
50 20 7	**12.1**	12.1	14.3	14.7	26.6	**0**	9e3	**12.1**	12.1	21	21.3	30.8	**0**	1e4	**12.1**	12.1	**11.9**	**12.2**	**21.3**	**0**	9e4	**12.1**	**11.8**	18.1	18.1	30.3	**0**	**9e4**
70 10 3	**7.5**	6.7	7.9	9.1	19.5	1	1e3	**7.5**	6.4	9.1	10	19.3	1	8e2	**7.5**	6.6	**6**	**7.1**	19.3	1	**1e4**	**7.5**	**5.5**	7.1	**7.1**	**18.8**	**2**	7e3
70 10 7	**15.2**	14.6	20.1	20.4	32.7	**0**	1e3	**15.2**	14.6	23	23.3	33.2	**0**	1e3	**15.2**	**14.5**	**14.4**	**14.5**	**26.9**	**0**	1e4	**15.2**	**14.5**	21.2	21.3	33.2	**0**	**1e4**
70 20 3	**1.1**	1.1	0.9	1.3	3.8	0	5e2	**1.1**	0.9	8.3	8.5	28.4	2	2e3	**1.1**	1	**0.7**	**1**	**2.6**	1	5e3	**1.1**	**0.7**	6.3	6.3	28.4	**4**	**3e4**
70 20 5	**4.6**	**4.6**	5.2	5.3	13.8	**0**	6e2	**4.6**	**4.6**	7.7	7.8	24.3	**0**	5e3	**4.6**	**4.6**	**4.6**	**4.6**	**10.7**	**0**	7e3	**4.6**	**4.6**	6.5	6.5	24.3	**0**	**5e4**
	F_p^{km2} 0.5h							F_z^{km2} 0.5h							F_p^{km2} 5h							F_z^{km2} 5h						

Table 5
BMSTP results comparison for the best formulations.

$\|V\|$ d $\|C\|$	$objL$	$objU$	gUL	t	$objL$	$objU$	gUL	t	$objL$	$objU$	gUL	t
20 30 7	**541**	597	9.38	-	**541**	**541**	-	**101.6**	**541**	556	2.7	-
20 50 3	**190**	**190**	-	**0**	**190**	**190**	-	**0**	**190**	**190**	-	**0**
20 50 5	395	467	15.42	-	**407**	467	12.85	-	**407**	**414**	**1.69**	-
30 30 3	**413**	425	2.82	-	**413**	**413**	-	947.1	**413**	**413**	-	**150.1**
30 50 3	**1830**	1862	1.72	-	**1830**	**1830**	-	19.7	**1830**	**1830**	-	**9.9**
30 50 5	1254	**1320**	5	-	**1320**	**1320**	-	**31.3**	1188	**1320**	10	-
30 50 7	497	524	5.15	-	**506**	524	3.44	-	**506**	**506**	-	**1511.9**
50 10 5	**1470**	1588	7.43	-	**1470**	1528.6	3.83	-	**1470**	**1470**	-	**3686.1**
50 10 7	732	828	11.59	-	**734**	**769.2**	**4.58**	-	**734**	778.6	5.73	-
50 20 3	**2239**	2301	2.69	-	**2239**	2301	2.69	-	**2239**	**2239**	-	**2068.3**
50 20 7	582	**760**	23.42	-	**683**	799	**14.52**	-	641	795	19.37	-
70 10 3	4599	4694	2.02	-	**4641**	**4641**	-	6409.9	**4641**	**4641**	-	**5878.6**
70 10 7	1604	**2002**	19.88	-	**1787**	2023	**11.67**	-	1646	2023	18.64	-
70 20 3	759	**763**	0.52	-	**763**	**763**	-	**245.6**	**763**	**763**	-	388.6
70 20 5	934	**1173**	20.38	-	**1086**	**1173**	**7.42**	-	1019	**1173**	13.13	-
70 30 5	1167	**1227**	4.89	-	**1227**	**1227**	-	**500.7**	1083	**1227**	11.74	-
	F_{mor} 5h				F_p^{km2} 5h				F_z^{km2} 5h			